

## Follow the Sun, 1 Axis, Program Description

The program uses a real time clock for time keeping. The motion of the array is controlled by 4 time settings, defining when moves occur. The move times are expressed as minutes after midnight. For example one is called EightAM, and is set to  $8*60=480$ .

The 1st move, at 8AM, is 45 degrees to the east. The second move is defined by a setting, say NineThirtyAM, and it is the first of 24 moves to the west, ending at ThreePM. The array movement dwells for 2 hours, before returning to horizontal at FivePM.

The hardware consists of 4 components. 1.) An Arduino UNO clone. 2.) An LCD display and 5 button keypad from DFRobot. This board plugs directly into the UNO. 3.) A solid state relay board from SainSmart. This board gets Vcc and Ground from the UNO. The board has 2 relays, which connect to D2 and D3 on the LCD board. 4.) A Real Time Clock board. This board has a button Lithium Cell that keeps the clock running when power is absent. It communicates to the UNO thru an I2C connection. The actuator motor (110 volts AC) is one that rotates 72 RPM, then thru a 9:1 reduction. It could be a DC actuator at 12 volts. In this case, the relay is a different style.

### Main and Subroutine description:

1. Main is a while 1 loop with a switch statement. The three elements of the switch statement are Stop, Run, and Set Time. Choose the Up, Down, or Right pushbutton to select Run or Stop or Set Time.

2. Subroutines -

Version - Displays software version upon startup.

Instructions – Displays operating instructions.

Timer 1 Overflow - The Real Time Clock maintains the time over the long haul. The hour and minute are updated daily. But over a 24 hour period, an interrupt is used to maintain the minute. The interrupt is called Timer 1 Overflow. The target chip clock speed is 15,999,552 Hz. This is divided by 8. When a 16 bit register fills 1831 times, a minute variable is incremented. But this method is inaccurate to the tune of 2 minutes per 24 hours. Hence, the real time clock is used, as its accuracy is better than 1 minute per year.

Move1Axis - This switches 110 volts AC to the actuator motor. 2 relays are used. One sets the motor direction; the other applies power to the motor. There are 3 wires to the motor. See the schematic.

Button Press - an interrupt is called when a button press occurs. The routine sets a variable that determines where to be in the switch statement, or in the case of setting the time, increments the variables. Currently this is hours and minutes. In the future, a 2 axis software package will use the day of the year.

Run - The Run subroutine is divided into 2 sections. Tracking, or stored horizontally when not tracking.

MinutesToHour Minute - converts a global variable Minutes to Hours:Minutes.

PowerFromBattery - The real time clock can be set to where it gets its power from the button battery. This does that.

GetAndDisplayTime – Obtains the time from the real time clock, at 12:01, each day.

SetTime - Does that to the real time clock. This occurs once or twice during the lifetime of the system, when the Lithium button cell needs to be replaced.

A note about time keeping. In version one of the single axis controller, time was approximated by using a light sensor to count dark minutes during the night time. Minutes, as measured from midnight, was simply  $\text{Dark minutes} / 2$  plus a correction factor. Version 2 of the controller uses an accurate real time clock (RTC).

The code has been composed with the aid of Flowcode – [www.matrixtls.com](http://www.matrixtls.com)